I. OBJECTION TO THE DRAWINGS

The drawings have been objected to because of unclear labeling. The Examiner suggests that, with respect to FIGURES 6 and 7, to label the FIGURE 6A and 6B, and 7A and 7B rather than FIGURE 6 and FIGURE 6 (cont'd) and FIGURE 7 and FIGURE 7 (cont'd). The Applicants have adopted the Examiner's suggestion and are filing concurrently herewith corrected drawings. Additionally, the brief description of FIGURES 6 and 7 have been rewritten hereinabove, accordingly.

II. OBJECTION TO THE SPECIFICATION

The Specification has been objected to because Application serial numbers are missing in the Cross-Reference to Related Applications. The Cross-Reference to Related Applications has accordingly been amended hereinabove. Additionally, the Applicants have amended the written description to correct a minor typographical error with respect to the reference numeral of GHV0.

III. OBJECTION TO THE CLAIMS

Claim 2 has been objected to because the claim refers to a "second value" without a reference to an "first value." Claim 2 has been cancelled hereinabove.

Claim 19 has been objected to because of a typographical error in the word "claim." Claim 19 has been cancelled hereinabove.

IV. REJECTION UNDER 35 U.S.C. § 103

Claims 1-6, 8, 9 and 11-18 have been rejected as being unpatentable over *Patt, et al.*, "Alternative Implementations of Hybrid Branch Predictors," Proceedings of the 28th

Annual Symposium on Microarchitecture, 1995, pp. 252-257 ("*Patt*"). The Applicants respectfully traverse the rejection of claims 1-6, 8, 9 and 11-18 under 35 U.S.C. § 103.

Claims 6, 8, 9 and 11-18 have been cancelled hereinabove without prejudice or disclaimer. Claims 1-6, 8, 9 and 11-18 are respectfully withdrawn from the Examiner's consideration.

V. REJECTION UNDER 35 U.S.C. § 103

Claims 7, 10, 19 and 20 have been rejected under 35 U.S.C. § 103 as being unpatentable over *Patt* in view of *Hennessy, et al.*, "Computer Architecture: A Quantitive Approach", p. 269 ("*Hennessy*"). The Applicant respectfully traverses the rejection of claims 7, 10, 19 and 20 under 35 U.S.C. § 103.

Claims 7 and 19 and 20 have been cancelled hereinabove without prejudice or disclaimer. Claims 7, 19 and 20 are respectfully withdrawn from the Examiner's consideration.

With respect to claim 10 has been rewritten hereinabove in independent form. Claim 10 is directed to a processing system including a first branch history table comprising a plurality of bimodal accessed entries for storing a first set of branch prediction bits, a second branch history table comprising a plurality of fetch-based accessed entries for storing a second set of branch prediction bits, a selector for selecting in response to a selection control bit selected from a set of selection control bits, a bit from a selected one of said sets of bits accessed from said first and second branch history tables, and a selector table comprising a plurality of entries for storing said set of selector bits as a function of a performance history of said first and second sets of branch prediction bits stored in said first and second branch history tables, wherein each said entry in said tables comprises a one-bit counter. Claim 10 has been rejected over the combination of *Patt* and *Hennessy*. *Hennessy* is relied on as teaching *n*-bit predictors, in which *n* could be one. (Paper No. 3, page 11.) *Patt* admittedly does not disclose the

limitation in which each entry in the tables comprises a one-bit counter. (Paper No. 3, page 11.)

To the contrary, *Patt* teaches, in particular, a Branch Predictor Selection Table (BPST) having two-bit counters. (*Patt*, Section 2, page 252.) The teaching in *Patt* with respect to such a BPST discloses hybrid branch prediction schemes in which the counters in the BPST keep track of the currently more accurate predictor for a branch. (*Id.*) The two-level prediction scheme disclosed in *Patt* is a combination of the hybrid scheme with a first level history based on branch history registers (BHRs). (*Patt*, Section 4.1, page 255.) *Patt* explicitly teaches that the BPST used in the two-level prediction scheme is a table of two-bit counters just as disclosed in conjunction with the hybrid prediction scheme (*Patt*, Section 4.2, pages 255-56.) As discussed in conjunction therewith, as noted hereinabove, the two-bit counters of the BPST keep track of the predictor which is currently more accurate for a particular branch.

It is contended that it would have been obvious to use one-bit counters in *Patt* to have the advantage of simple circuitry. (Paper No. 3, page 11.) The Applicant respectfully disagrees for several reasons. A motivation to combine or modify a reference must be found in one of three possible sources thereof. MPEP § 2143.01. These are the teachings of the reference or references, the knowledge of persons of ordinary skill in the art or the nature of the problem to be solved. *Id.* Furthermore, the teachings of a motivation or suggestion must be clear and particular. *In re Lee*, 277 F.3d 1338, 1343, 61 U.S.P.Q.2d 1430, 1433-34 (Fed. Cir. 2002); *In re Kotzab*, 217 F.3d 1365, 1371, 55 U.S.P.Q.2d 1313, 1317 (Fed. Cir. 2000); *In re Dembiczak*, 175 F.3d 994, 999, 50 U.S.P.Q.2d 1614, 1616 (Fed. Cir. 1999). Broad conclusory statements regarding the reference or references are not evidence. *In re Kotzab*, 217 F.3d at 1371, 55 U.S.P.Q.2d at 1317; *In re Dembiczak*, 175 F.3d at 999, 50 U.S.P.Q.2d at 1616. Consequently, a broad recitation that the one-bit counters lead to simpler circuitry is insufficient to support a *prima facie* showing of obviousness.

Moreover, the performance of the two-level scheme of *Patt* depends on, *inter alia*, the number of bits in the counters of the BPST; *Patt* discloses that the counters of

the BPST are incremented or decremented depending on which of the single-scheme predictors constituting the hybrid branch predictor was correct. (*Patt*, Section 2, page 252.) Therefore, the number of bits in the counters of the BPST provide a history, of which predictor is the more accurate predictor of the branch. (*See, e.g., Patt*, Section 2, page 252; Section 4.1, page 255.) In other words, the two-bit counters constitute the second level history in the two-level scheme of *Patt*. (*Patt*, Section 4.1, page 255.) Consequently, modifying the two-bit counter BPST of *Patt* to use a one-bit counter changes the principle operation of *Patt*. Thus, there can be no motivation or suggestion to modify *Patt* to make the invention of claim 10. MPEP § 2143.01. Furthermore, there must be some reasonable expectation of success in making the modification. MPEP § 2143.02. The reasonable expectation of success must be found in the prior art. MPEP § 2143. In the instant case, there can be no reasonable expectation of success because modifying *Patt* to replace the two-bit counters of the BPST reduces the second level of history in the two-level branch predictor scheme of *Patt*. This would be expected to adversely affect the figure of merit, that is, the misprediction rate in *Patt*.

Thus, for at least the aforesaid reasons, the Applicants respectfully assert that it is not *prima facie* obvious to modify *Patt* to incorporate the teachings of *Hennessy* to make the invention of claim 10. Therefore, claim 10 is allowable under 35 U.S.C. § 103 over *Patt* and *Hennessy*.

V. NEW CLAIMS 21-40

Claims 21-40 have been added to more particularly point and distinctly claim that which the Applicants regard as the invention.

New claim 21 is directed to branch prediction circuitry including a bimodal branch history table comprising a plurality of entries, each restoring a prediction value and access by selective bits of a branch address; a fetch-based branch history table comprising a plurality of entries for storing a prediction value and accessed by a pointer generated from selected bits of the branch address and bits from a history register, each entry of the fetch-based branch history table operable for containing bits representing a

prediction value for a plurality of branches in a fetch group; and a selector table comprising a plurality of entries each restoring a plurality of selection bits and accessed by a pointer generated from selected bits from said branch address and bits from said history register, each said selector bit used for selecting between a bimodal prediction value accessed from the bimodal history table and a prediction value accessed from the fetch-based history table. *Patt* does not disclose, at least, a fetch-based branch history table including entries operable for containing bits representing a prediction value for a plurality of branches in a fetch group. Support for new claim 21 may be found in the Detailed Description at, *inter alia*, page 10, line 13 through page 11, line 4.

New claim 22 depends from claim 21 and further recites circuitry for updating the bimodal and fetch-based branch history tables operable to set a corresponding entry in each of the bimodal and fetch-based branch history tables to a first value when a branch is taken at branch resolution time, and set a corresponding entry in each of the bimodal and fetch-based branch history tables to a second value when a branch is not taken at branch resolution time. New claim 22 parallels original claim 2.

New claim 23 recites the branch prediction circuitry of claim 21 in which the history register comprises a shift register and the branch prediction circuitry further comprises circuitry for updating the shift register by shifting in a preselected prediction value for each fetch group. Support for claim 23 may be found in the Detailed Description, page 28, lines 16-17. *Patt* does not, at least, disclose updating a history factor by shifting in a value for each fetch group.

New claim 24 recites a branch prediction circuitry of claim 21 and further includes limitations paralleling those of original claim 3.

New claim 25 recites the branch prediction circuitry of claim 21 in which the plurality of selection bits are operable for selecting a first subset of prediction values from the bimodal branch history table and a second subset of prediction values from the fetch-based branch history table. Support for claim 25 may be found in the Detailed Description at page 11, lines 1-5. *Patt* does not disclose, at least, a plurality of selection

bits operable for selecting a first subset of prediction values from a bimodal branch history table and a second subset of prediction values from the fetch-based branch history table.

New claim 26 recites the branch prediction circuitry of claim 23 including limitations paralleling those in original claims 4 and 5.

New claim 27 is directed to the branch prediction circuitry of claim 23 and recites additional limitation paralleling that of original claim 6. *Patt* admittedly does not teach or suggest the limitation of claim 6. (Paper No. 4, page 6.) On the contrary, *Patt* discloses leaving the counter state in the BPST unchanged if both prediction schemes were incorrect. (*Patt*, Section 2, page 252.) Consequently, the Applicant respectfully submits that the assertion that it would have been obvious to record the fetch-based table as the one the selector should choose, as it is more likely to be accurate, is without objective evidence or support, and moreover, *Patt* teaches away from such combination. (*See* Paper No. 3, page 6.)

New claim 28 is directed to a processing system including a first branch history table comprising a plurality of bimodally accessed entries, each entry restoring a first set of branch prediction bits; a second branch history table comprising a plurality of fetch-based access entries, each entry restoring a second set of branch prediction bits; a selector for selecting, in response to a plurality of selection control bits, a set of prediction bits in a selected one of said sets of bits accessed from said first and second branch history table; and a selector table comprising a plurality of entries, each entry for storing a plurality of selection control bits wherein the selection control bits are set as a function of performance history of corresponding first and second sets of branch prediction bits stored in the first and second branch history tables. The support for new claim 28 may be found, *inter alia*, at page 9, line 13, through page 11, line 9, and the accompanying drawings. *Patt* does not teach, at least, first and second branch history tables including entries for storing corresponding sets of branch prediction bits, a selector for selecting a set of prediction bits... and a selector table having entries for storing a plurality of selection control bits.

New claim 29 recites the system of claim 28 further including the limitation paralleling the limitation of original claim 9. *Patt* does not teach, at least, fetch-based accessing.

New claim 30 recites the processing system of claim 28 and further including the limitation paralleling the limitation of original claim 10.

New claims 31, 32 and 33 depend, respectively from claim 28, 31 and 32, and recite the limitations paralleling original claims 11, 12 and 13, respectively.

New claim 34 is directed to a method of performing branch predictions in a processing system including a bimodal branch history table, a fetch-based branch history table and a selector table. The method includes the substeps of accessing the bimodal branch history table to retrieve a first set of branch prediction bits; accessing the fetch-based branch history table to retrieve a set of second branch prediction bits; selecting between the first and second sets of branch prediction bits in response to corresponding bits retrieved from the selector table, wherein a sum of a number of bits in the first set of branch prediction bits and a number of bits in the second set of branch prediction bits is not less than a number of instructions in a fetch group; and updating the selector table as a function of actual branch resolution. Support for new claim 34 may be found in the Detailed Description at page 10, line 13 through page 11, line 4. *Patt* does not teach, at least, accessing the bimodal and fetch-based branch history tables to retrieve corresponding sets of branch prediction bits, or a number of the bits in the first set of branch prediction bits and a number of bits in the second set of branch prediction bits, sum to at least a number of instructions in a fetch group.

New claim 35 depends from claim 34 and recites the additional limitation in which the step of updating the selector table includes the substeps of determining if at least one of the first set of branch prediction bits correctly predicts the corresponding branch resolution outcome; updating the corresponding entry in the selector table to a first logic value when the at least one of the first sets of prediction bits correctly represents the branch resolution outcome; determining if at least one of the second set of

branch prediction bits correctly predicts the branch resolution outcome; and updating the corresponding entry in the selector table to a second logic value when the at least one of the second branch prediction bits correctly represents the branch resolution outcome. Support for new claim 35 may be found in the Detailed Description at page 22, line 20, through page 23, line 25. *Patt* necessarily does not disclose updating the selector table including the substeps of claim 35 for at least the reason that *Patt* does not teach the first and second set of branch prediction bits as recited therein.

New claim 36 depends from claim 35 and recites the additional steps of determining if at least one bit of both the first and second sets of branch history correctly predicts the branch resolution outcome; maintaining the current value of corresponding bits and the corresponding selector table when the at least one bit of both the first and second sets of branch prediction bits correctly predict the branch resolution outcome; determining if at least one bit of both the first and second sets of branch prediction that incorrectly predicts the resolution outcome; and maintaining the current value of corresponding bits in the corresponding selector table entry when the at least one bit of both the first and second sets of branch history bits incorrectly predict the branch history outcome. The support for new claim 36 may also be found in the aforementioned portion of the Detailed Description. For at least the same reasons as discussed in conjunction with claim 35, *Patt* does not teach or suggest the limitations of claim 36.

New claim 37 depends from claim 35 which cites the additional steps of determining whether at least one bit of both the first and second sets of branch prediction bits correctly predict the branch resolution outcome; maintaining the current value of corresponding bits in the corresponding selector table when the at least one bit of both the first and second sets of branch prediction bits correctly predict the branch resolution outcome; and updating the current selector table entry to a logic value associated with a fetch-based branch history table when neither of corresponding bits of the first and second sets of branch prediction bits correctly predict the branch resolution outcome. The support for new claim 37 may be found in the aforementioned portion of the Detailed Description discussed in conjunction with claim 35. For at least the reasons discussed

hereinabove in conjunction with claim 27, *Patt* does not teach or suggest the limitations of claim 37.

New claim 38 recites the method of claim 34 in which the step of accessing the fetch-based branch history table comprises the substep of generating an address from at least some bits of a branching instruction and bits received from a history register. The limitation of claim 38 parallel the limitation of original claim 18.

New claim 39 depends from claim 38 and recites the limitation in which the history register comprises a shift register. The limitation of claim 39 parallels the limitation of original claim 19.

New claim 40 depends from claim 39 in which the method further comprises the steps of updating the shift register by shifting in a prediction bit for each fetch group. The limitation of new claim 40 parallels the limitation of new claim 23 with respect to shifting in a preselected prediction value for each fetch group.

VII. CONCLUSION

As a result of the foregoing, it is asserted by Applicants that the remaining claims in the Application are in condition for allowance, and respectfully request an early allowance of such claims.

Applicants respectfully request that the Examiner call Applicants' attorney at the below listed number if the Examiner believes that such a discussion would be helpful in resolving any remaining problems.

Respectfully submitted,

WINSTEAD SECHREST & MINICK P.C.
Attorney for Applicants

By: Barry S. Newberger
Barry S. Newberger
Reg. No. 41,527

5400 Renaissance Tower
1201 Elm Street
Dallas, Texas 75270
(512) 370-2808

VERSION TO SHOW CHANGES MADE

IN THE CROSS-REFERENCE TO RELATED APPLICATIONS

The cross-reference to related applications, lines 2-9, page 1, has been rewritten as follows:

The present invention is related to the following U.S. Patent Application filed concurrently herewith and hereby incorporated herein by reference:

Serial No. 09/434,856 (Attorney Docket No. AT9-98-536) entitled "Apparatus and Method for Controlling Link Stack Corruption During Speculative Instruction Branching;" and

Serial No. 09/434,763 (Attorney Docket No. AUS990815US1) entitled "Apparatus and Method for Accessing a Memory Device During Speculative Instruction Branching."

IN THE BRIEF DESCRIPTION OF THE DRAWINGS

(1) The brief description of FIGURE 6 has been rewritten as follows:

FIGURE 6 (including partial views FIGURE 6A and FIGURE 6B) is a flow diagram illustrating a first method of updating the local and fetch-based branch history and selector tables of FIGURE 3A; and

(2) The brief description of FIGURE 7 has been rewritten as follows:

FIGURE 7 (including partial views FIGURE 7A and FIGURE 7B) is a flow diagram illustrating a second method of updating the local and fetch-based branch history and selector tables of FIGURE 3A.

IN THE DETAILED DESCRIPTION

(1) The paragraph beginning at line 3, page 13 has been rewritten as follows:

Refer now to FIGURE 4 illustrating GHV logic 311 in further detail. The current value of GHV, which is loaded into GHV register 306, FIGURE 3A, is provided from GHV0 logic [402] 401. GHV0 logic 401 may be a register having a two-way multiplexer input. A first input to the multiplexer portion is coupled to an output of multiplexer (MUX) 414. Multiplexer 414 selects for outputting a GHV value from one of several paths that are rendered active in response to an event, cache miss, branch misprediction, pipeline hold, etc. that may modify the value of one GHV. Each of these paths and the operation of GHV logic 311 will be described in detail below.

IN THE CLAIMS

Claim 10 has been rewritten as follows:

10. (Amended) [The processing system of Claim 8] A processing system comprising:

a first branch history table comprising a plurality of bimodally accessed entries for storing a first set of branch prediction bits;

a second branch history table comprising a plurality of fetch-based accessed entries for storing a second set of branch prediction bits;

a selector for selecting in response to a selection control bit selected from a set of selection control bits, a bit from a selected one of said sets of bits accessed from said first and second branch history tables; and

a selector table comprising a plurality of entries for storing said a set of selector bits as a function of a performance history of said first and second sets of branch prediction bits stored in said first and second branch history tables, wherein said each said entry in said tables comprises a 1-bit counter.